

**PEMROGRAMAN BERORIENTASI OBYEK**  
**(JURNAL 03 – KELAS & OBYEK)**



**Disusun oleh:**

Muhamad Dzikriansyah  
607062300103

**D3 Rekayasa Perangkat Lunak Aplikasi**  
**Fakultas Ilmu Terapan**  
**Universitas Telkom**  
**2024**

## Soal 1

Class Buku:

```
public class Buku {  
    private String judul;  
    private String pengarang;  
    private double harga;  
  
    // constructor  
    public Buku(String judul, String pengarang, double harga) {  
        this.judul = judul;  
        this.pengarang = pengarang;  
        this.harga = harga;  
    }  
  
    // setter  
    public void setJudul(String judul) {  
        this.judul = judul;  
    }  
  
    public void setPengarang(String pengarang) {  
        this.pengarang = pengarang;  
    }  
  
    public void setHarga(double harga) {  
        this.harga = harga;  
    }  
  
    // getter  
    public String getJudul() {  
        return judul;  
    }  
  
    public String getPengarang() {  
        return pengarang;  
    }  
  
    public double getHarga() {  
        return harga;  
    }  
}
```

Class Main:

```
public class MainBuku {  
    public static void main(String[] args) {  
  
        // instantiating object
```

```

        Buku buku1 = new Buku("Harry Potter", "J.K. Rowling", 300000);
        Buku buku2 = new Buku("UML", "Ivar Jacobson", 400000);

        System.out.println("Judul buku1: " + buku1.getJudul());
        System.out.println("Pengarang buku1: " + buku1.getPengarang());
        System.out.println("Harga buku1: " + buku1.getHarga());

        System.out.println();

        System.out.println("Judul buku2: " + buku2.getJudul());
        System.out.println("Pengarang buku2: " + buku2.getPengarang());
        System.out.println("Harga buku2: " + buku2.getHarga());
    }
}

```

Output:

```

PS D:\school D3 RPLA 47-04\SEM 3\PBO
e' '-XX:+ShowCodeDetailsInExceptionMe
49\redhat.java\jdt_ws\Jurnal 3 Kelas
Judul buku1: Harry Potter
Pengarang buku1: J.K. Rowling
Harga buku1: 300000.0

Judul buku2: UML
Pengarang buku2: Ivar Jacobson
Harga buku2: 400000.0
PS D:\school D3 RPLA 47-04\SEM 3\PBO

```

Soal 2

Class Pegawai:

```

public class Pegawai {
    private final String idPegawai;
    private String nama;
    private String alamat;
    private String departemen;
    private double gaji;

    public Pegawai(String id, String nama, String alamat, String departemen, double
gaji) {
        this.idPegawai = id;
        this.nama = nama;
        this. alamat = alamat;
        this.departemen = departemen;
        this.gaji = gaji;
    }

    // setter

```

```
public void setNama(String nama) {
    this.nama = nama;
}

public void setAlamat(String alamat) {
    this.alamat = alamat;
}

public void setDepartemen(String departemen) {
    this.departemen = departemen;
}

public void setGaji(double gaji) {
    this.gaji = gaji;
}

// getter
public String getId() {
    return idPegawai;
}

public String getNama() {
    return nama;
}

public String getAlamat() {
    return alamat;
}

public String getDepartemen() {
    return departemen;
}

public double getGaji() {
    return gaji;
}

@Override
public String toString() {
    return "ID Pegawai: " + idPegawai +
        "\nNama: " + nama +
        "\nAlamat: " + alamat +
        "\nDepartemen: " + departemen +
        "\nGaji: " + gaji + "\n";
}
```

```
}
```

Class Kantor:

```
import java.util.ArrayList;

public class Kantor {
    ArrayList<Pegawai> listPegawai;

    // constructor
    public Kantor() {
        listPegawai = new ArrayList<>();
    }

    // function
    public void addPegawai(Pegawai p) {
        listPegawai.add(p);
    }

    public void getPegawai() {
        for (Pegawai p : listPegawai) {
            System.out.println(p);
        }
    }

    public void getPegawai(String departemen) {
        for (Pegawai p : listPegawai) {
            if (p.getDepartemen().equalsIgnoreCase(departemen)) {          //
                //
                p.getDepartemen() returns string | .equalsIgnoreCase() is string method
                System.out.println(p);
            }
        }
    }

    public void getGajiPegawai(double gaji) {
        for (Pegawai p : listPegawai) {
            if (p.getGaji() > gaji) {
                System.out.println(p);
            }
        }
    }

    public void setDataPegawai(String id, String nama, String alamat, String departemen,
double gaji) {
        for (Pegawai p : listPegawai) {
            if (p.getId().equals(id)) {
                p.setNama(nama);
                p.setAlamat(alamat);
                p.setDepartemen(departemen);
                p.setGaji(gaji);
            }
        }
    }
}
```

```

    }
}

}

}

```

Class Main:

```

import java.util.Scanner;

public class MainPegawai {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);

        Kantor listPegawaiKantor = new Kantor();

        int choice = 0;

        do {
            System.out.println("\n--Data Pegawai--" +
                "\n1. Input Pegawai" +
                "\n2. Edit Data Pegawai" +
                "\n3. Tampilkan Seluruh Data Pegawai" +
                "\n4. Tampilkan Data Pegawai Berdasarkan Departemen" +
                "\n5. Tampilkan Data Pegawai Berdasarkan Gaji" +
                "\n0. Keluar");

            System.out.print("Pilih opsi: ");
            choice = s.nextInt();
            s.nextLine();

            switch (choice) {
                case 1:
                    System.out.println();
                    System.out.print("Masukkan id pegawai: ");
                    String id = s.nextLine();
                    System.out.print("Masukkan nama pegawai: ");
                    String nama = s.nextLine();
                    System.out.print("Masukkan alamat: ");
                    String alamat = s.nextLine();
                    System.out.print("Masukkan departemen: ");
                    String departemen = s.nextLine();
                    System.out.print("Masukkan gaji: ");
                    double gaji = s.nextDouble();
                    s.nextLine();

```

```

        Pegawai p = new Pegawai(id, nama, alamat, departemen, gaji);

        listPegawaiKantor.addPegawai(p);

        System.out.println("-- Data pegawai berhasil ditambahkan --");

        break;
    case 2:
        System.out.println();
        System.out.print("Masukkan id pegawai yang ingin diubah: ");
        String idPeg = s.nextLine();

        System.out.print("Masukkan nama pegawai: ");
        nama = s.nextLine();
        System.out.print("Masukkan alamat: ");
        alamat = s.nextLine();
        System.out.print("Masukkan departemen: ");
        departemen = s.nextLine();
        System.out.print("Masukkan gaji: ");
        gaji = s.nextDouble();
        s.nextLine();

        listPegawaiKantor.setDataPegawai(idPeg, nama, alamat, departemen,
gaji);

        System.out.println("-- Pegawai dengan id: " + idPeg + " berhasil
diubah --");

        break;
    case 3:
        System.out.println("\n-- Data Pegawai --");
        listPegawaiKantor.getPegawai();
        break;
    case 4:
        System.out.print("Masukkan nama departemen: ");
        departemen = s.nextLine();

        System.out.println("\n-- Data Pegawai Departemen " + departemen);
        listPegawaiKantor.getPegawai(departemen);
        break;
    case 5:
        System.out.print("Masukkan nominal gaji: ");
        gaji = s.nextDouble();

        System.out.println("\n-- Data pegawai dengan gaji di atas " + gaji +
" --");

        listPegawaiKantor.getGajiPegawai(gaji);
        break;

```

```

        default:
            System.out.println("Input tidak valid");
        }

    } while (choice != 0);

}
}

```

### Opsi 1: Memasukkan data

```

HK Pemrograman Berorientasi Objek\Jurnal 3 Kelas &
sages' '-cp' 'C:\Users\dzikr\AppData\Roaming\Code\l
Objek_51586448\bin' 'MainPegawai'

```

```

--Data Pegawai--
1. Input Pegawai
2. Edit Data Pegawai
3. Tampilkan Seluruh Data Pegawai
4. Tampilkan Data Pegawai Berdasarkan Departemen
5. Tampilkan Data Pegawai Berdasarkan Gaji
0. Keluar
Pilih opsi: 1

```

```

Masukkan id pegawai: A01
Masukkan nama pegawai: Dzikri
Masukkan alamat: Bandung
Masukkan departemen: SDM
Masukkan gaji: 9000000
-- Data pegawai berhasil ditambahkan --

```

### Opsi 2: Mengubah data berdasarkan id

```

--Data Pegawai--
1. Input Pegawai
2. Edit Data Pegawai
3. Tampilkan Seluruh Data Pegawai
4. Tampilkan Data Pegawai Berdasarkan Departemen
5. Tampilkan Data Pegawai Berdasarkan Gaji
0. Keluar
Pilih opsi: 2

```

```

Masukkan id pegawai yang ingin diubah: A01
Masukkan nama pegawai: Dzikri
Masukkan alamat: Bandung
Masukkan departemen: SDM
Masukkan gaji: 9500000
-- Pegawai dengan id: A01 berhasil diubah --

```



### Opsi 3: Menampilkan data seluruh pegawai

```
--Data Pegawai--
1. Input Pegawai
2. Edit Data Pegawai
3. Tampilkan Seluruh Data Pegawai
4. Tampilkan Data Pegawai Berdasarkan Departemen
5. Tampilkan Data Pegawai Berdasarkan Gaji
0. Keluar
Pilih opsi: 3
```

```
-- Data Pegawai --
ID Pegawai: C03
Nama: John
Alamat: Bandung
Departemen: Keuangan
Gaji: 8000000.0
```

```
ID Pegawai: A02
Nama: Turah
Alamat: Bali
Departemen: SDM
Gaji: 7000000.0
```

```
ID Pegawai: B03
Nama: Yogi
Alamat: Sukapura
Departemen: Produksi
Gaji: 5000000.0
```

```
ID Pegawai: A04
Nama: Dany
Alamat: Buahbatu
Departemen: SDM
Gaji: 8000000.0
```

```
ID Pegawai: B05
Nama: Elif
Alamat: Bojongsoang
Departemen: Keuangan
Gaji: 5000000.0
```

```
ID Pegawai: A01
Nama: Dzikri
Alamat: Bandung
Departemen: SDM
Gaji: 9500000.0
```

### Opsi 4: Menampilkan data berdasar kan departemen

```
--Data Pegawai--
1. Input Pegawai
2. Edit Data Pegawai
3. Tampilkan Seluruh Data Pegawai
4. Tampilkan Data Pegawai Berdasarkan Departemen
5. Tampilkan Data Pegawai Berdasarkan Gaji
0. Keluar
Pilih opsi: 4
Masukkan nama departemen: SDM
```

```
-- Data Pegawai Departemen SDM
ID Pegawai: A02
Nama: Turah
Alamat: Bali
Departemen: SDM
Gaji: 7000000.0
```

```
ID Pegawai: A04
Nama: Dany
Alamat: Buahbatu
Departemen: SDM
Gaji: 8000000.0
```

```
ID Pegawai: A01
Nama: Dzikri
Alamat: Bandung
Departemen: SDM
Gaji: 9500000.0
```

## Opsi 5: Menampilkan data berdasar range gaji

```
--Data Pegawai--
1. Input Pegawai
2. Edit Data Pegawai
3. Tampilkan Seluruh Data Pegawai
4. Tampilkan Data Pegawai Berdasarkan Departemen
5. Tampilkan Data Pegawai Berdasarkan Gaji
0. Keluar
Pilih opsi: 5
Masukkan nominal gaji: 5000000

-- Data pegawai dengan gaji di atas 5000000.0 --
ID Pegawai: C03
Nama: John
Alamat: Bandung
Departemen: Keuangan
Gaji: 8000000.0

ID Pegawai: A02
Nama: Turah
Alamat: Bali
Departemen: SDM
Gaji: 7000000.0

ID Pegawai: A04
Nama: Dany
Alamat: Buahbatu
Departemen: SDM
Gaji: 8000000.0

ID Pegawai: A01
Nama: Dzikri
Alamat: Bandung
Departemen: SDM
Gaji: 9500000.0
```